```
FFFFFFFFFFFFFFF     000000000     RRRRRRRRRRRR     RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
FFFFFFFFFFFFFFF     000000000     RRRRRRRRRRRR     RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
FFFFFFFFFFFFFFF     000000000     RRRRRRRRRRRR     RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
FFF               000         000 RRR         RRR RRR         RRR       TTT        LLL
FFF               000         000 RRR         RRR RRR         RRR       TTT        LLL
FFF               000         000 RRR         RRR RRR         RRR       TTT        LLL
FFF               000         000 RRR         RRR RRR         RRR       TTT        LLL
FFF               000         000 RRR         RRR RRR         RRR       TTT        LLL
FFF               000         000 RRR         RRR RRR         RRR       TTT        LLL
FFFFFFFFFFFF      000         000 RRRRRRRRRRRR    RRRRRRRRRRRR          TTT        LLL
FFFFFFFFFFFF      000         000 RRRRRRRRRRRR    RRRRRRRRRRRR          TTT        LLL
FFFFFFFFFFFF      000         000 RRRRRRRRRRRR    RRRRRRRRRRRR          TTT        LLL
FFF               000         000 RRR    RRR      RRR    RRR            TTT        LLL
FFF               000         000 RRR    RRR      RRR    RRR            TTT        LLL
FFF               000         000 RRR    RRR      RRR    RRR            TTT        LLL
FFF               000         000 RRR        RRR  RRR        RRR        TTT        LLL
FFF               000         000 RRR        RRR  RRR        RRR        TTT        LLL
FFF               000         000 RRR        RRR  RRR        RRR        TTT        LLL
FFF                 000000000     RRR            RRR RRR            RRR   TTT        LLLLLLLLLLLLLLL
FFF                 000000000     RRR            RRR RRR            RRR   TTT        LLLLLLLLLLLLLLL
FFF                 000000000     RRR            RRR RRR            RRR   TTT        LLLLLLLLLLLLLLL
```

```
FFFFFFFFFF    000000    RRRRRRRR    PPPPPPPP      AAAAAA    RRRRRRRR
FFFFFFFFFF    000000    RRRRRRRR    PPPPPPPP      AAAAAA    RRRRRRRR
FF          00      00  RR      RR  PP      PP   AA      AA  RR      RR
FF          00      00  RR      RR  PP      PP   AA      AA  RR      RR
FF          00      00  RR      RR  PP      PP   AA      AA  RR      RR
FF          00      00  RR      RR  PP      PP   AA      AA  RR      RR
FFFFFFFF    00      00  RRRRRRRR    PPPPPPPP     AA      AA  RRRRRRRR
FFFFFFFF    00      00  RRRRRRRR    PPPPPPPP     AA      AA  RRRRRRRR
FF          00      00  RR  RR      PP           AAAAAAAAAA  RR  RR
FF          00      00  RR  RR      PP           AAAAAAAAAA  RR  RR
FF          00      00  RR    RR    PP           AA      AA  RR    RR        ....
FF          00      00  RR    RR    PP           AA      AA  RR    RR        ....
FF            000000    RR      RR  PP           AA      AA  RR      RR      ....
FF            000000    RR      RR  PP           AA      AA  RR      RR      ....

  SSSSSSSS  DDDDDDDD    LL
  SSSSSSSS  DDDDDDDD    LL
SS          DD      DD  LL
SS          DD      DD  LL
SS          DD      DD  LL
SS          DD      DD  LL
  SSSSSS    DD      DD  LL
  SSSSSS    DD      DD  LL
        SS  DD      DD  LL
        SS  DD      DD  LL
        SS  DD      DD  LL
        SS  DD      DD  LL
SSSSSSSS    DDDDDDDD    LLLLLLLLLL
SSSSSSSS    DDDDDDDD    LLLLLLLLLL
```

```
{ REQUIRE file for FORTRAN internal parameters passed between modules
{ File: FORPAR.SDL  Edit: SBL2001
{**************************************************************************
{*                                                                        *
{*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                               *
{*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                *
{*  ALL RIGHTS RESERVED.                                                  *
{*                                                                        *
{*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
{*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE *
{*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER *
{*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
{*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY *
{*  TRANSFERRED.                                                          *
{*                                                                        *
{*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE *
{*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT *
{*  CORPORATION.                                                          *
{*                                                                        *
{*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS *
{*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.               *
{*                                                                        *
{*                                                                        *
{**************************************************************************
{
{
{ Author:  T. Hastings
{ Change History:
{ [Previous edit history removed.  SBL 23-Aug-1982]
{ 1-007 - Add FOR$K_CLASS_SB and FOR$K_CLASS_NL.  JAW 28-Jul-1981
{ 2-001 - Convert to SDL.  SBL 23-Aug-1982

{ Define all symbols as if they were globals using the FOR$ prefix.
{ Call definitions in MACRO-32 with:
{       $FORPAR


MODULE $FORPAR;

AGGREGATE FOR$R_PAR STRUCTURE PREFIX FOR$;

{+
{ Define statement type structure which is passed in R0 from UPI-level
{ statement entry routines to FOR$$IO_BEG.
{-

    union_1 UNION;
        structure_1 STRUCTURE;
            STMT_TYPE BYTE UNSIGNED;            { statement type code
            STMT_FLAGS BYTE UNSIGNED;           { statement flags
            END structure_1;
        structure_2 STRUCTURE;
            fill_1 BITFIELD LENGTH 8 FILL TAG $$; { skip statement type
            OBJ_FMT BITFIELD;                   { run-time (object time) formatting
            END structure_2;
        END union_1;
```

```
{+
{ Private-use descriptor class codes defined for Fortran.  These codes
{ identify descriptors passed by compiled code to FOR$IO_ELEM.
{-

    CONSTANT CLASS_SB EQUALS 191;
    CONSTANT CLASS_NL EQUALS 190;

{+
{ Control Z record for use with reading ENDFILE produced on PDP-11s
{-

    CONSTANT CONTROL_Z EQUALS 26;       { ASCII Control Z - ENDFILE record

{+
{ Parameter values to be passed to Handlers set up
{ on I/O calls from user (see FORERROR.B32).
{ The values indicate what cleanup action is to be done when
{ and if an UNWIND should occur (ERR= transfer or SIGNAL or STOP
{ which does an UNWIND.
{-

    CONSTANT UNWINDPOP EQUALS 0;        { UNWIND cleanup which pops current LUB/ISB/RAB
    CONSTANT UNWINDNOP EQUALS 1;        { UNWIND cleanup which does nothing (ie NOP).
    CONSTANT UNWINDRET EQUALS 2;        { UNWIND cleanup which does a $CLOSE and returns LUB/ISB/RAB

    END FOR$R_PAR;

END_MODULE $FORPAR;

{       End of file FORPAR.SDL
```